



schaltstelle
gemeinsam. selbstständig.

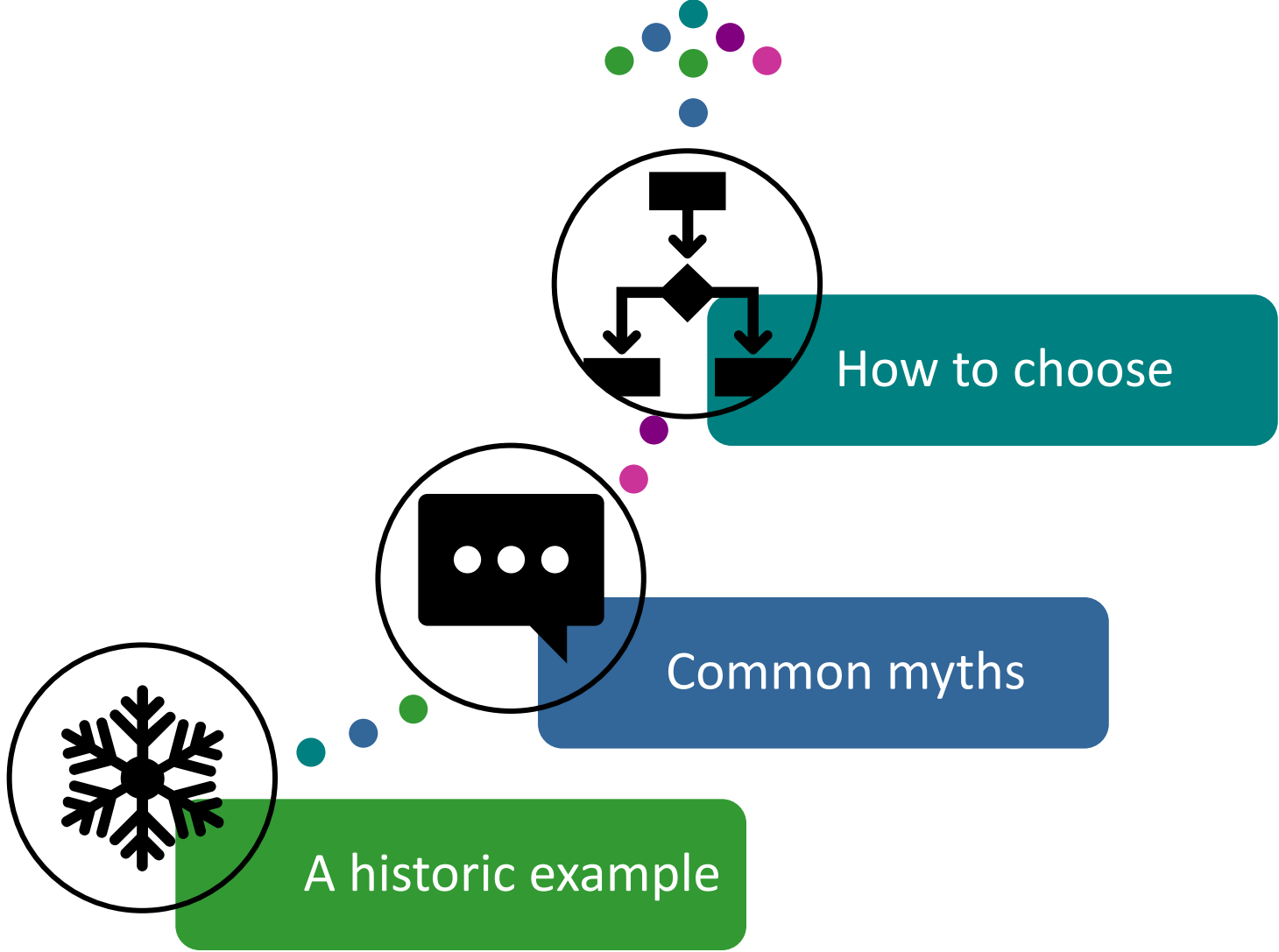


Myth Busting – NoSQL vs SQL data stores.

Or what database is good for your use case?

Jasmin Fluri

What does this talk cover? - Agenda



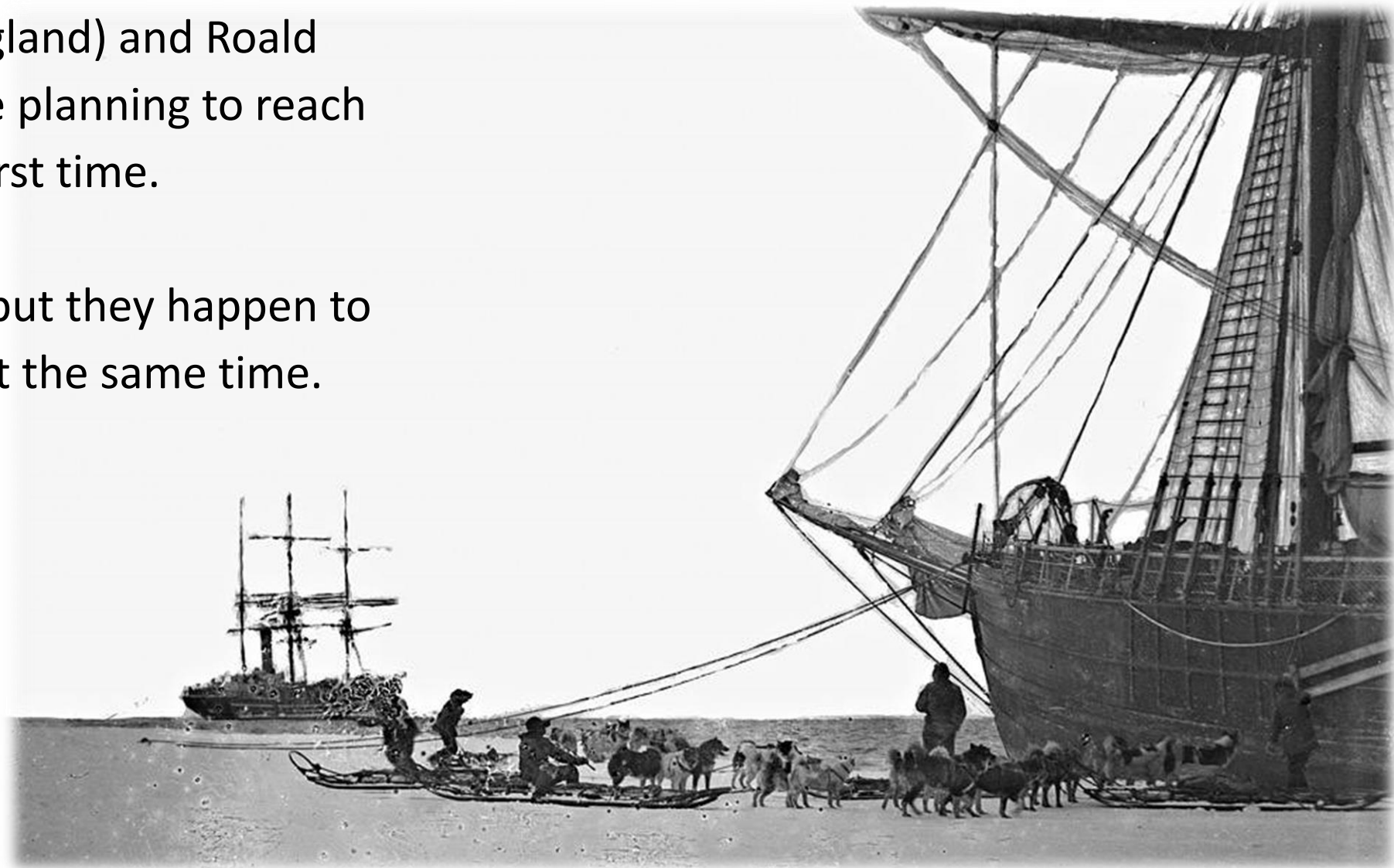
.... A short story about planning and technology and how it can affect a project outcome...



We go back to the year 1911...

Robert Falcon Scott (England) and Roald Amundsen (Norway) are planning to reach the south pole for the first time.

It's not a planned race, but they happen to start their expeditions at the same time.



schaltstelle

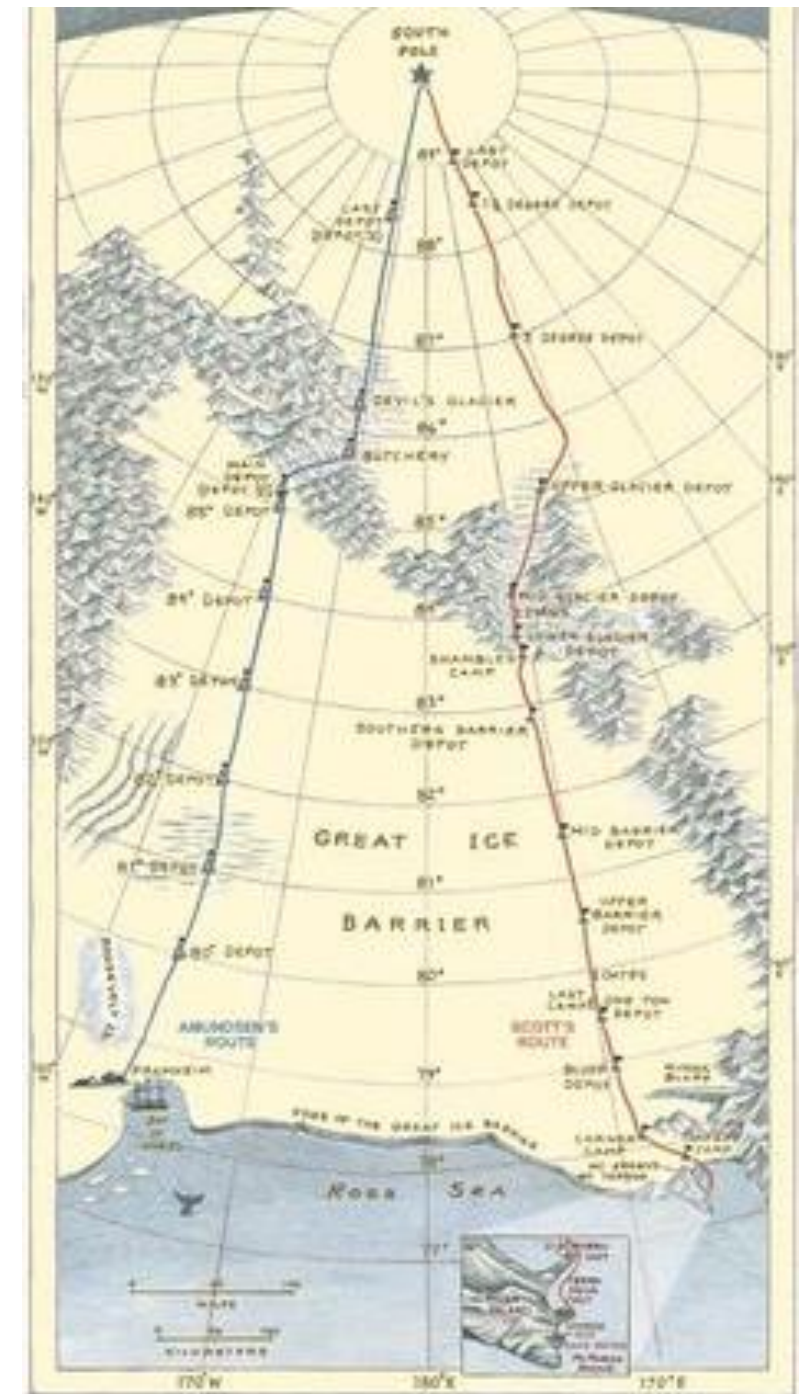
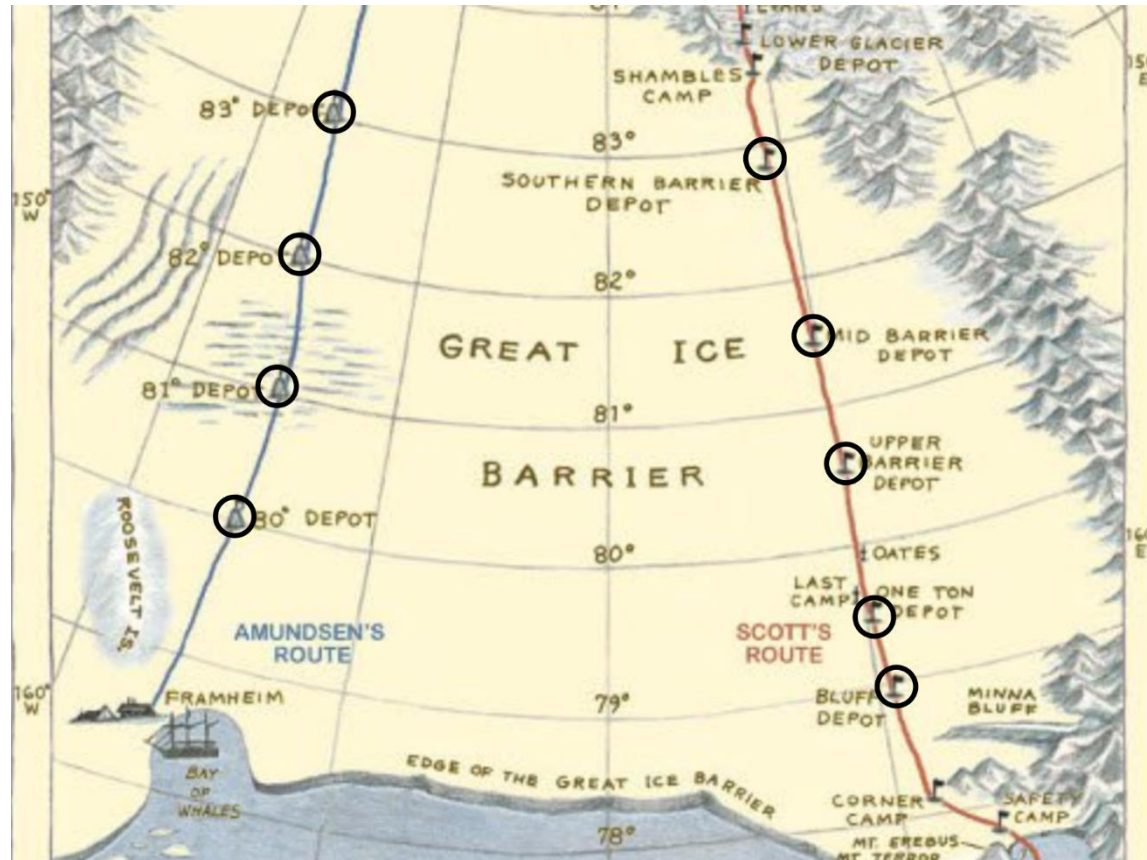
Different technologies

The same goal – reaching the south pole.

Problem: Logistics

Food depots had to be made to not carry all the baggage to the south pole and back to the bay.

- Fuel
- Food



What happened?



Amundsen starts at the Bay of Whales with a 60 miles advantage on 18 October 1911.



Scott starts at McMurdo Sound on November 1 1911
3 weeks after Amundsen.



Amundsen reaches the South Pole on 14 December 1911
33 days ahead of Scott!



Scott reached the South Pole on 17 January 1912, finding the Norwegian Flag.



Amundsen returns faster than expected his crew made it back in 99 days.



Scott and two of his men died on their way back due to exhaustion, hunger and extreme cold.



Which part played «choosing the right technology» in this story?



They chose to use very different technologies!
But had the same goal!

Housing

Amundsen Expedition

Tested tech



Scott Expedition

New, untested tech



Means of transport

Amundsen Expedition



Older, tested tech



Scott Expedition



New, untested tech

We'll be fine



schaltstelle

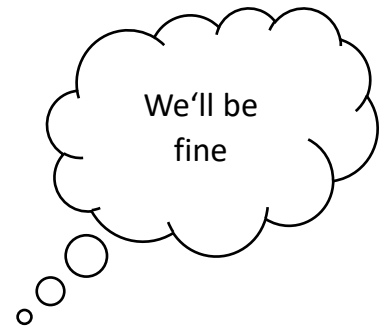
Planning vs. «We'll see when we get there»

Amundsen Expedition

Scott Expedition



Extra work for extra safety



We've got to be fast



Scott Polar Research Institute <http://www.spri.cam.ac.uk/>



schaltstelle

How do your technology choices affect your journey?



About me



Jasmin Fluri

Schaltstelle GmbH



@jasminfluri



@jasminfluri



jasminfluri



ORACLE
ACE

- Database Development
- Development Process
- Automation and Tooling
- Continuous Integration
- Continuous Delivery
- Software Engineering



schaltstelle

«Isn't that boring?»

«Do you also work with NoSQL?»



**Traditional
RDBMS**

Developers

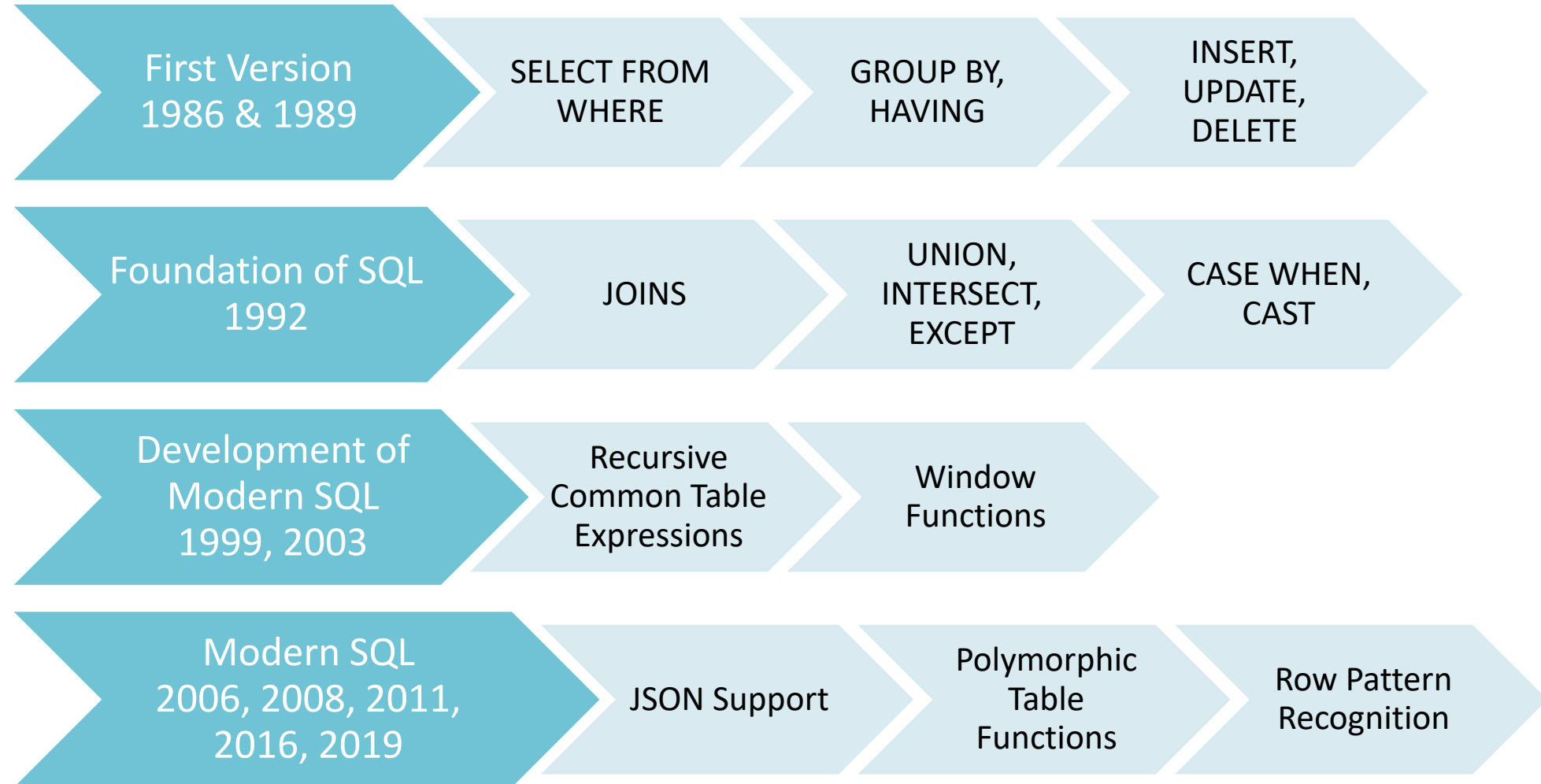
NoSQL DB

MYTH #1:

«Databases didn't improve over the last years... aren't they the same as 20 years ago?»



A (very short) History of the SQL Standard



MYTH #1:

«Databases didn't improve over the last years... aren't they the same as 20 years ago?»

BUSTED!



MYTH #2

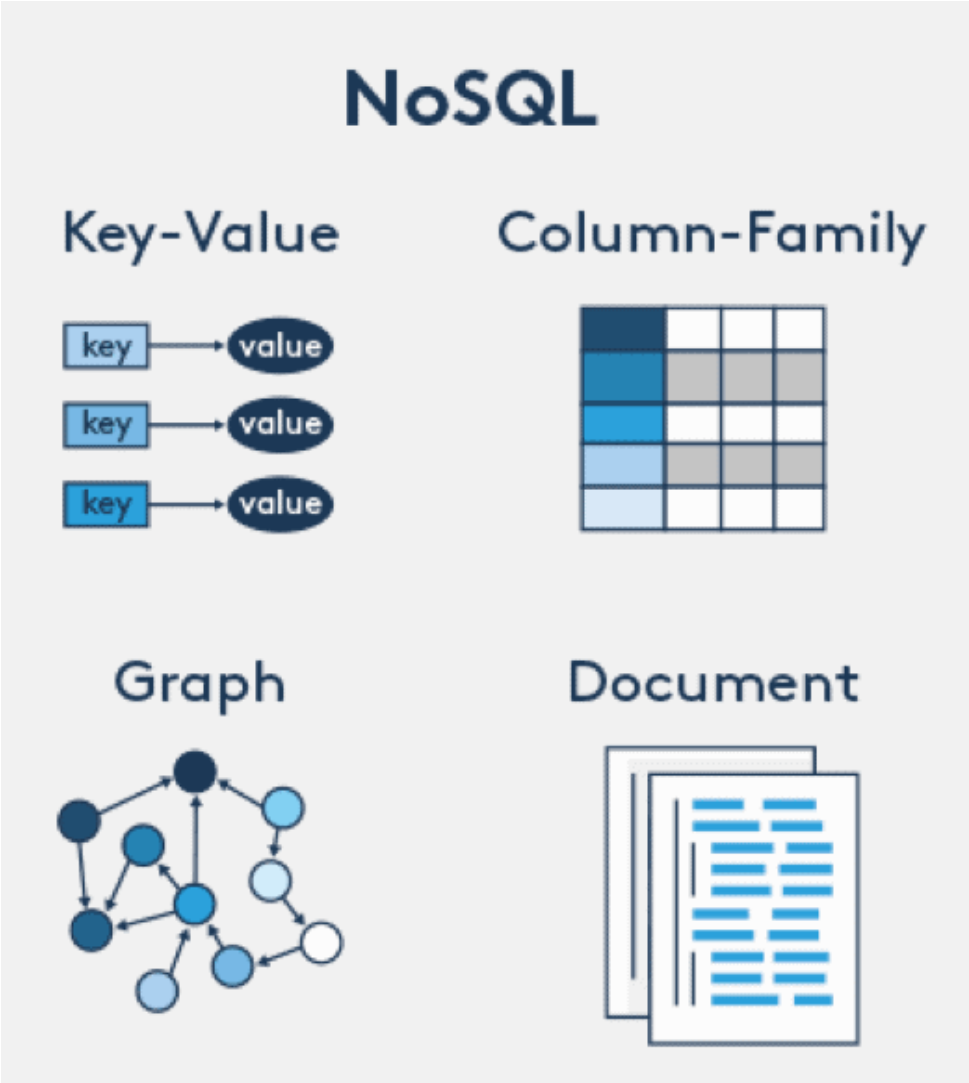
But we need NoSQL because our application stores data «unstructured».



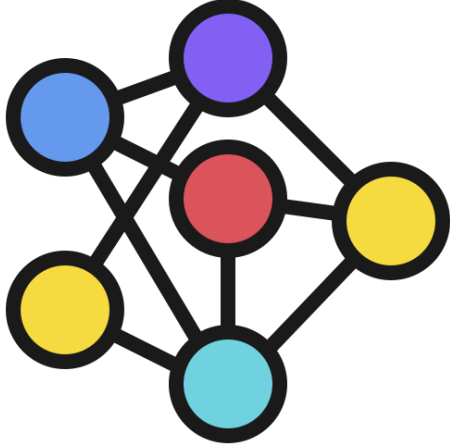
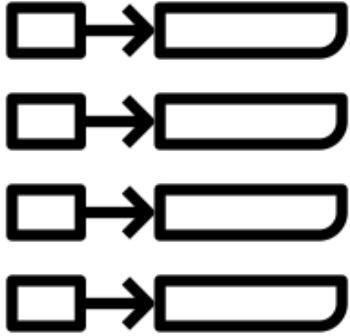
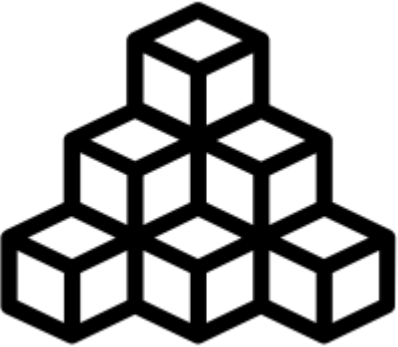
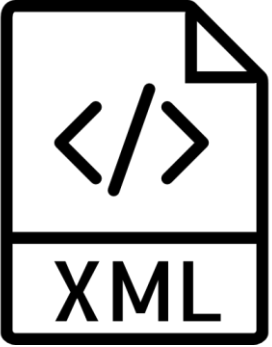
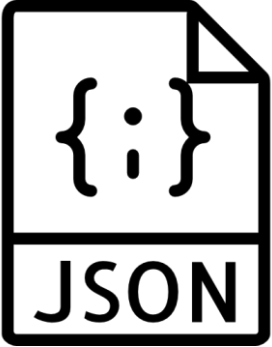
What is NoSQL?



What is NoSQL? – Is it really unstructured?



If your application can map your data, your data has a structure – the structure is just not relational!



MYTH #2

But we need NoSQL because our application stores data «unstructured».



MYTH #3

«But NoSQL data stores are a lot faster!»



Studies of NoSQL vs SQL performance

The challenge: finding a study that ...

- ... wasn't from a vendor.
- ... was executed on SSDs, because most NoSQL DBs are optimized for running on SSDs.
- ... was done in a clustered setup. Because most NoSQL DBs are optimized for running as a cluster and are scaling over multiple nodes.
- ... was done with more than one NoSQL technology.
- ... was done recently (not older than 5 years).



PERFORMANCE EVALUATION OF SQL AND NOSQL DATABASE MANAGEMENT SYSTEMS IN A CLUSTER

Christine Niyizamwiyitira and Lars Lundberg

Department of Computer Science and Engineering, Blekinge Institute of Technology, SE-37179 Karlskrona, Sweden.

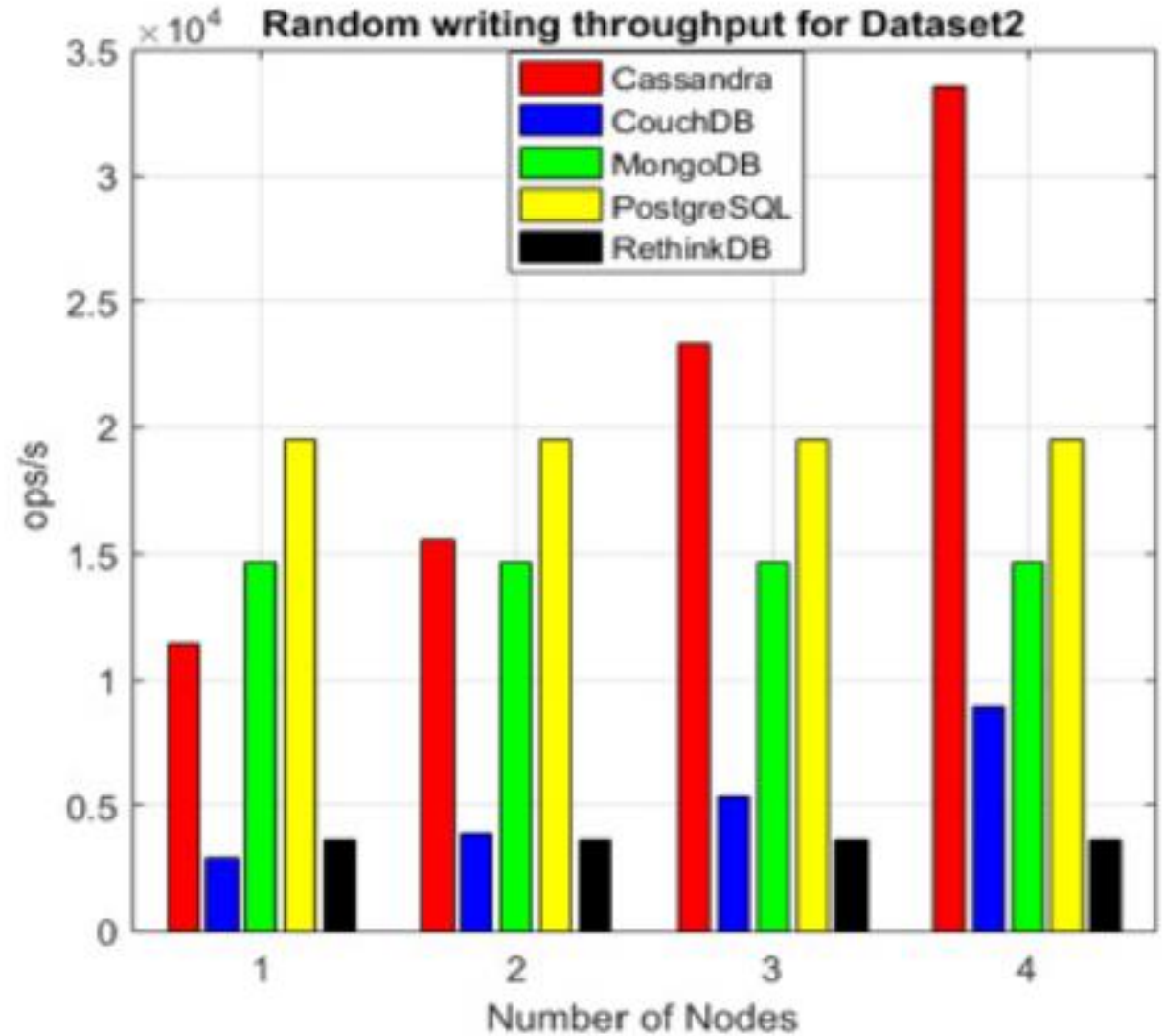
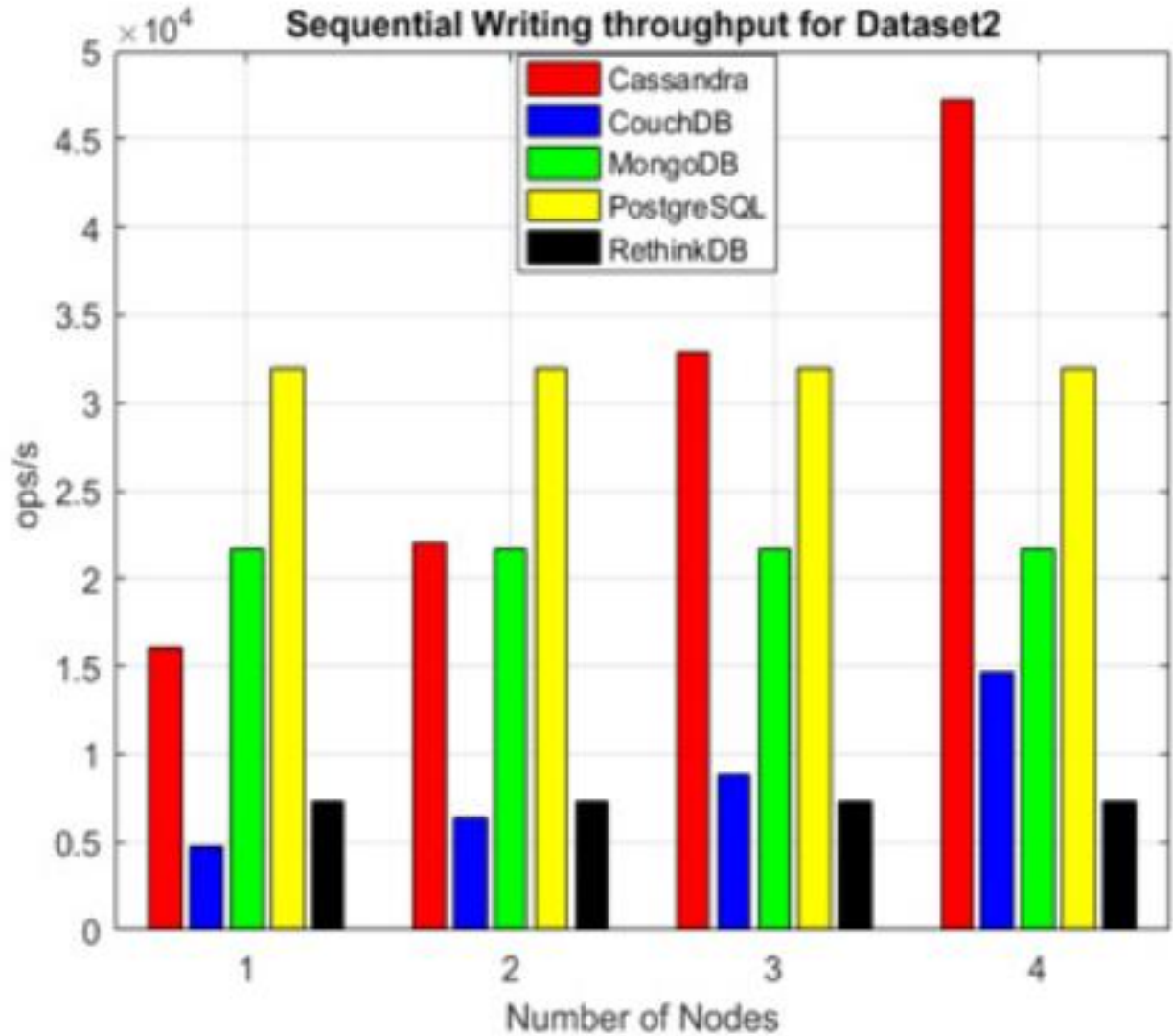
ABSTRACT

In this study, we evaluate the performance of SQL and NoSQL database management systems namely; Cassandra, CouchDB, MongoDB, PostgreSQL, and RethinkDB. We use a cluster of four nodes to run the database systems, with external load generators. The evaluation is conducted using data from Telenor Sverige, a telecommunication company that operates in Sweden. The experiments are conducted using three datasets of different sizes. The write throughput and latency as well as the read throughput and latency are evaluated for four queries; namely distance query, k-nearest neighbour query, range query, and region query. For write operations Cassandra has the highest throughput when multiple nodes are used, whereas PostgreSQL has the lowest latency and the highest throughput for a single node. For read operations MongoDB has the lowest latency for all queries. However, Cassandra has the highest throughput for reads. The throughput decreases as the dataset size increases for both write and read, for both sequential as well as random order access. However, this decrease is more significant for random read and write. In this study, we present the experience we had with these different database management systems including setup and configuration complexity.

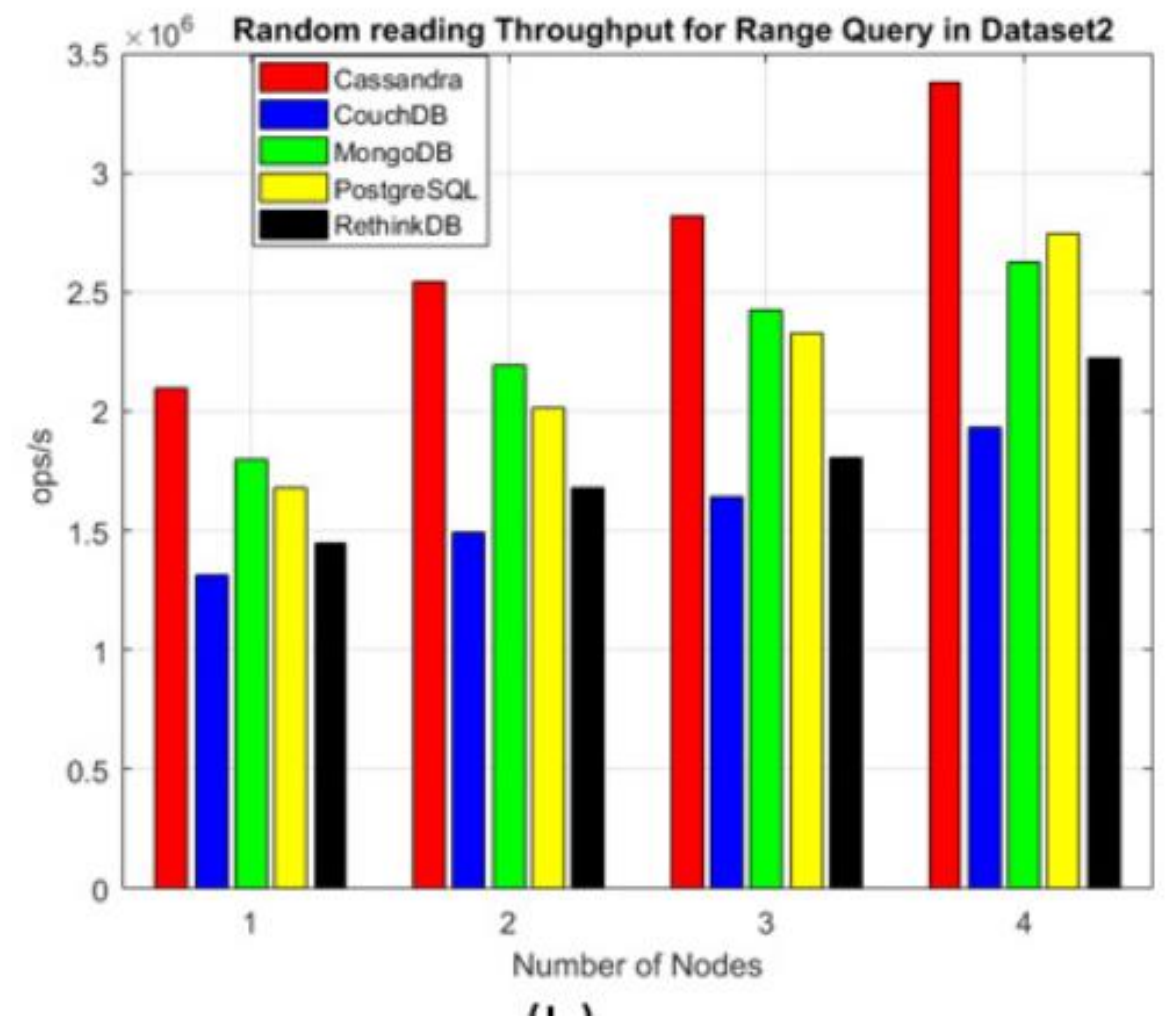
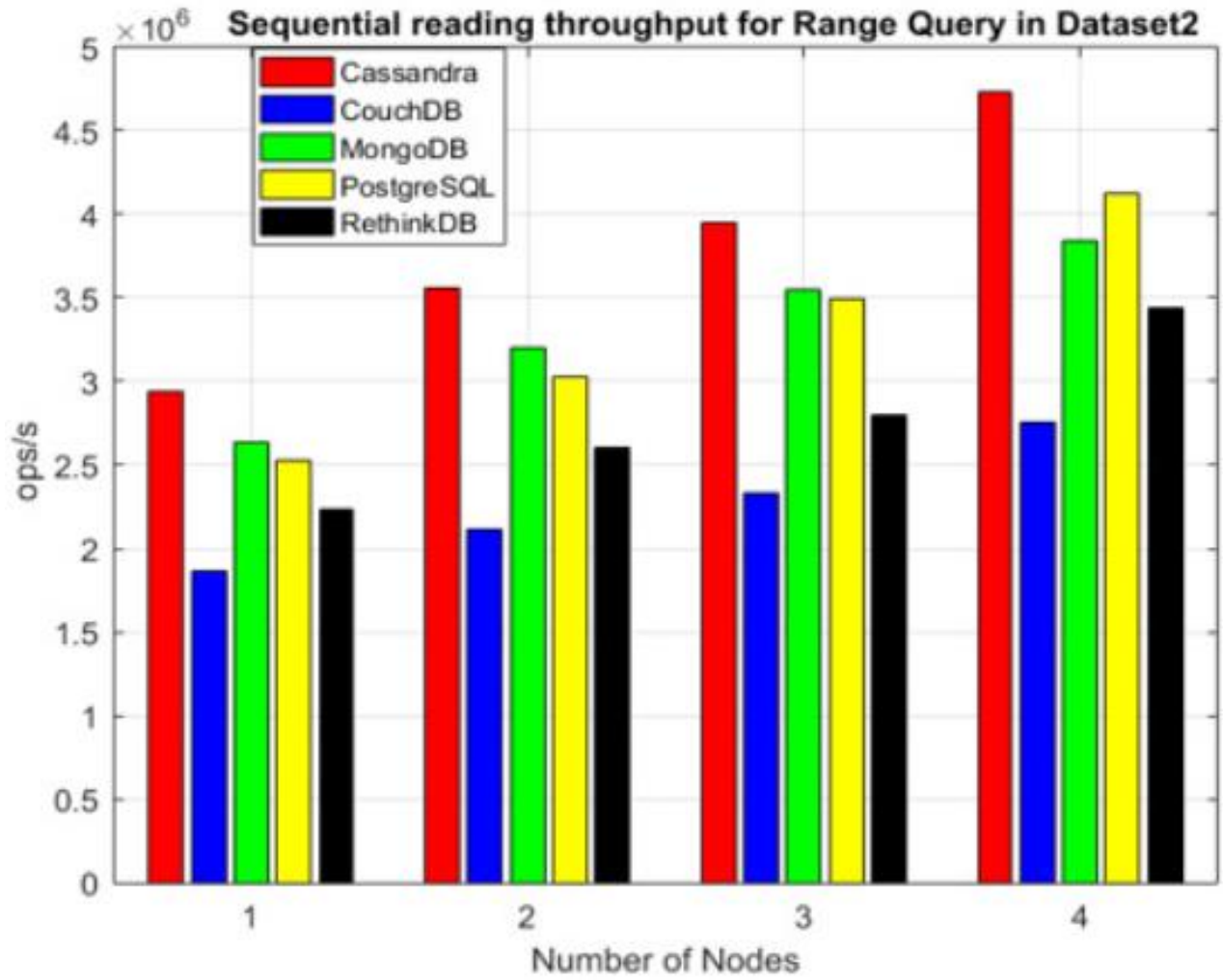
KEYWORDS

Trajectory queries, cluster computing, SQL database, NoSQL database, Cassandra, CouchDB, MongoDB, PostgreSQL, RethinkDB

NoSQL vs. SQL performance – Sequential vs. random writes



NoSQL vs SQL performance – Sequential vs random read for range query



schaltstelle

MYTH #3

«But NoSQL data stores are a lot faster!»

IT DEPENDS!



NoSQL DBs are faster than many SQL DBs in simple, read heavy scenarios.

APPROVED



SQL DBs are faster than
NoSQL DBs in complex read and
most heavy write
scenarios.

APPROVED



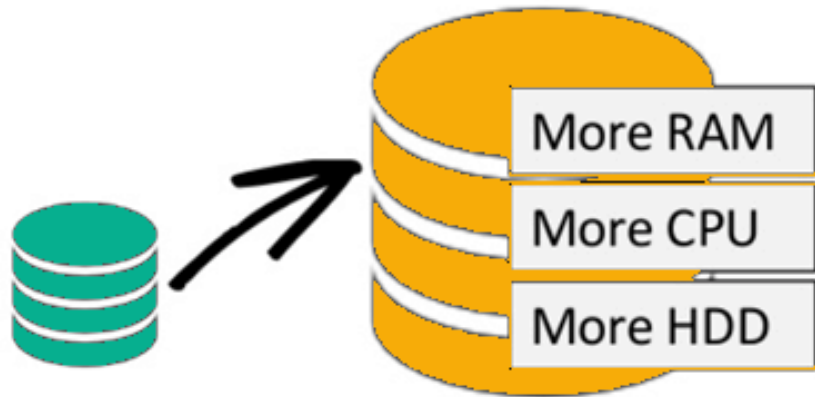
MYTH #4

«But relational databases don't scale!»

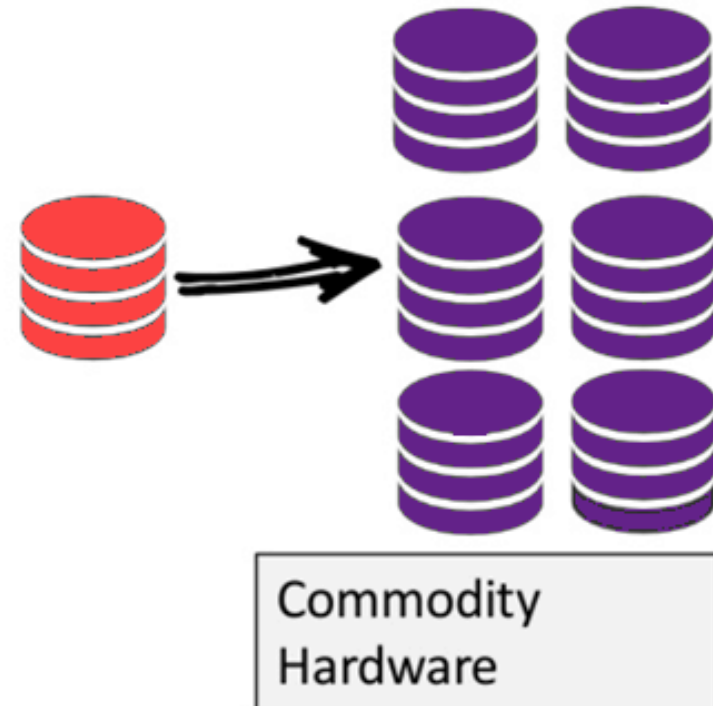


Two types of scaling....

Scale-Up (*vertical* scaling):



Scale-Out (*horizontal* scaling):



MYTH #4

«But relational databases don't scale!»

BUSTED!



Most relational Databases
don't scale horizontally!
They scale vertically!

APPROVED



Most NoSQL Databases
scale **horizontally!**

APPROVED



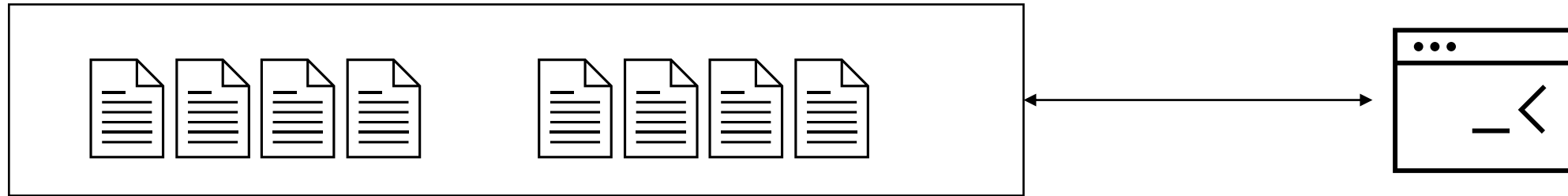
MYTH #5

«NoSQL DBs don't need schema migrations like relational data stores do!»

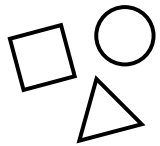


NoSQL data stores – schema changes

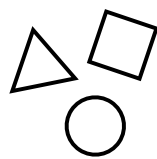
Collection



Schema
Version 1



Schema
Version 2



Can be solved with:

Query rewriting

(translating the current query into the old structure – can lead to higher latency)

Data migration

(transforming the old entries into the new structure)



NoSQL schema evolution

“If legacy entities are not migrated instantaneously in the event of a data model change, the new application code assumes a different data model than what is being stored in the production database.”



Self-adapting data migration in the context of schema evolution in NoSQL databases

[Andrea Hillenbrand](#) , [Uta Störl](#), [Shamil Nabiye](#) & [Meike Klettke](#)

[Distributed and Parallel Databases](#) (2021) | [Cite this article](#)

2281 Accesses | 2 Altmetric | [Metrics](#)

Abstract

When NoSQL database systems are used in an agile software development setting, data model changes occur frequently and thus, data is routinely stored in different versions. The management of versioned data leads to an overhead potentially impeding the software development. Several data migration strategies exist that handle legacy data differently during data accesses, each of which can be characterized by certain advantages and disadvantages. Depending on the requirements for the software application, we evaluate and compare different migration strategies through metrics like migration costs and latency as well as precision and recall. Ideally, exactly that strategy should be selected whose characteristics fulfill service-level agreements and match the migration scenario, which depends on the query workload and the changes in the data model which imply an evolution of the database schema. In this paper, we present a methodology of self-adapting data migration, which automatically adjusts migration strategies and their parameters with respect to the migration scenario and service-level agreements, thereby contributing to the self-management of database systems and supporting agile development.

MYTH #5

«NoSQL DBs don't need schema migrations like relational data stores do!»

BUSTED!



MYTH #6

«But developers are faster when using NoSQL stores!»



Facts about developer productivity!

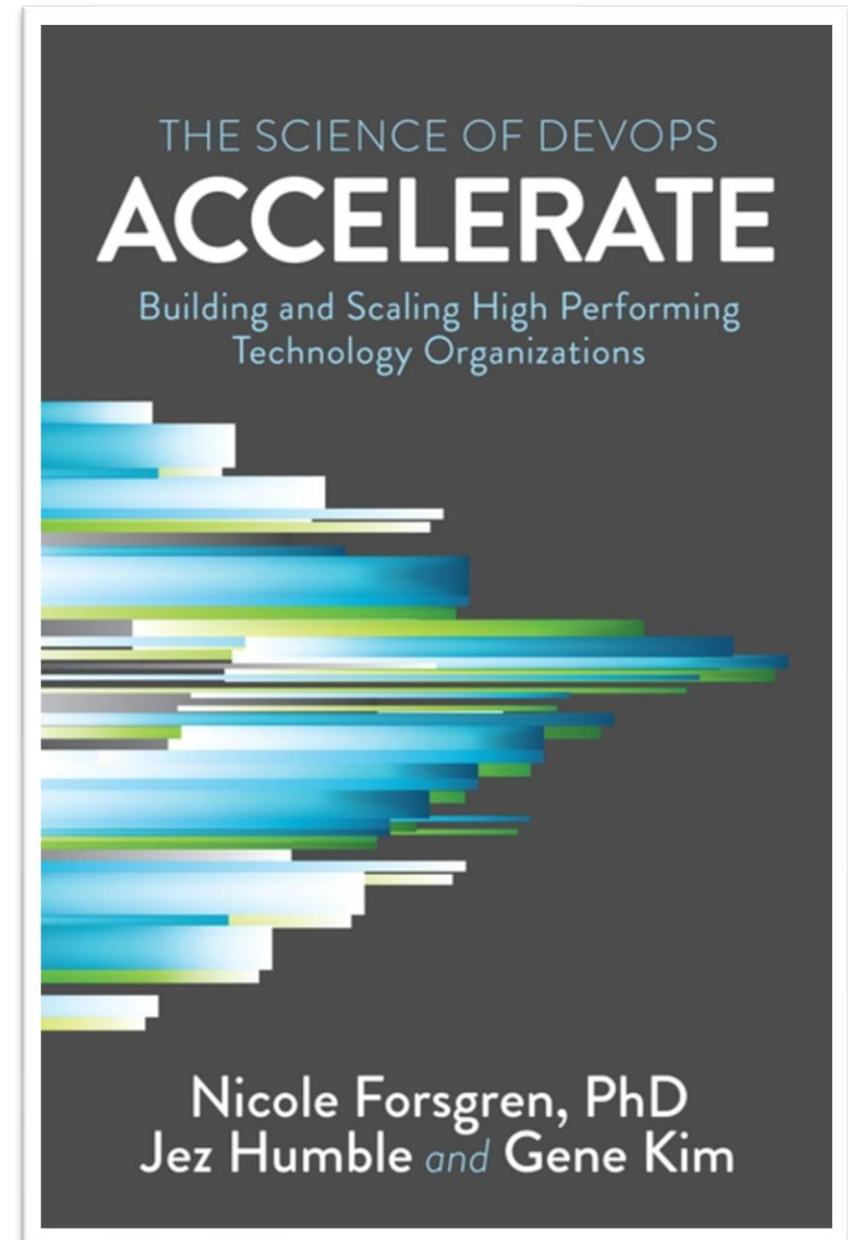
ACCELERATE

The Science of DevOps
Building and Scaling High Performing
Technology Organizations

by Nicole Forsgren, Jez Humble and Gene Kim



schaltstelle



«High performance is possible with all kind of systems and is independent of the technologies the teams use!»

Accelerate – The Science of DevOps



«Where code deployments are most painful, you'll find the poorest organizational performance.»

Accelerate – The Science of DevOps



MYTH #6

«But developers are faster when using NoSQL stores!»

BUSTED!




MYTH #7

«But we need Big Data storage to be flexible in application development!»

















Which Data Storage Technology does what?

Distributed File System
Object Storage



NoSQL DB

<p>Graph DB</p>    	<p>Key-Value Store</p>   	<p>Document Store</p>    
	<p>In-Memory Store</p>  	

Object-Relational Database



In-Memory Database



Search Index Database



How do you choose a data storage solution?



Do I already have a technology that solves my use case?

Yes



No



Can I find people who are willing to work with that technology?

Yes



No



If I have technical issues, how easy is it to find answers?

Yes



No



Is the technology choice obviously wrong for the task?

No



Yes



Is the technology enterprise ready?

Yes



No



MYTH #7

«But we need Big Data storage to be flexible in application development!»

BUSTED!



Some key take-aways...



Don't blindly choose the newest technology! – Choose the technology that fits your requirements best!



If «going the extra mile» gives you long-term advantages, you should do it!



Requirements will change!
Be prepared to have a plan B and
make sure your tools support
your plan B as well!



Summary - Blogpost

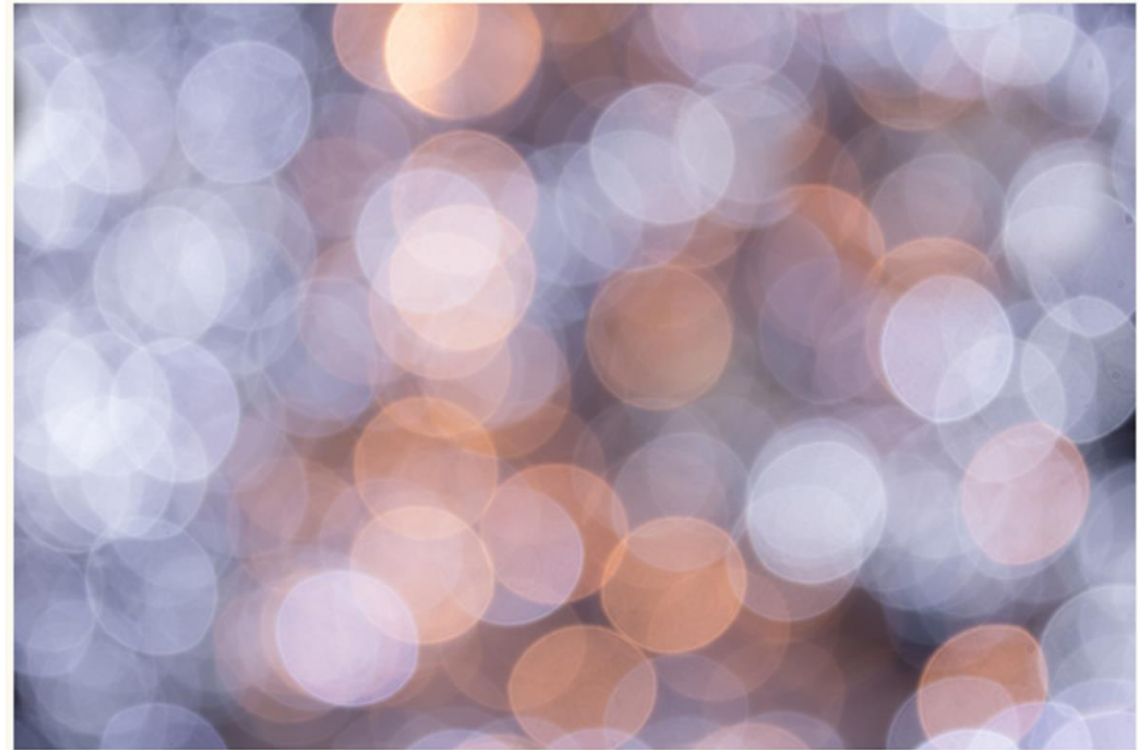


Foto von Pixabay von Pexels

Choose your new data storage solution with 5 questions!

When it comes to data storage solutions, there's no one size fits all. The data storage landscape offers a plethora of technologies. Some are single-purpose data stores; others are multi-purpose databases.



schaltstelle
gemeinsam. selbstständig.

Do you run a proper data storage solution
or is it a pony with snow shoes on?



@jasminfluri



@jasminfluri



jasminfluri

Sources / References

Single Purpose vs. Multi Purpose Databases – Blog

<https://blogs.oracle.com/database/many-single-purpose-databases-versus-a-converged-database>

Data Architecture – Inmon / Lindsted - Book

<https://www.sciencedirect.com/book/9780128020449/data-architecture-a-primer-for-the-data-scientist>

Data Migration in NoSQL Data Stores - Paper

<https://link.springer.com/article/10.1007/s10619-021-07334-1>

SQL Standard - Paper

<https://dl.acm.org/doi/10.1145/3299887.3299897>

Vertical vs. Horizontal Scaling – Blog

<https://blog.turbonomic.com/blog/on-technology/the-essentials-of-database-scalability-vertical-horizontal>

Performance Evaluation of SQL vs NoSQL Data Stores - Paper

<https://aircconline.com/ijdms/V9N6/9617ijdms01.pdf>



Sources / References

How to choose your new data storage solution with 5 questions - Blog

<https://medium.com/geekculture/choose-your-new-data-storage-solution-with-5-questions-f31308dd5e84>

Comparison of the Amundsen and Scott Expeditions

https://en.m.wikipedia.org/wiki/Comparison_of_the_Amundsen_and_Scott_expeditions

